**Project Design Phase-II**

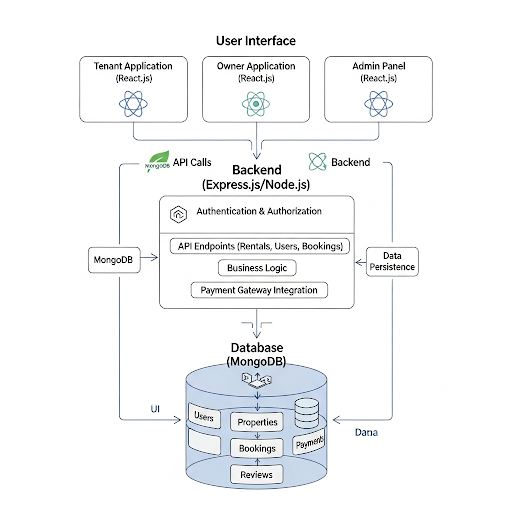
**Technology Stack (Architecture & Stack)**

|  |  |
| --- | --- |
| Date |  |
| Team ID | PNT2025TMID09657 |
| Project Name | Rentease-HOUSE RENT APP USING MERN |
| Maximum Marks | 4 Marks |

**Technical Architecture:**

The technical architecture of the house rental application is built on the **MERN stack**, which consists of four key components:

1. **Frontend (React.js):** This is the user interface that tenants, owners, and admins see and interact with. It's responsible for presenting property listings, application forms, and dashboards in a user-friendly way.
2. **Backend (Express.js & Node.js):** This is the server-side part of the application. It handles all the business logic, API calls, and user authentication. It acts as the bridge between the frontend and the database.
3. **Database (MongoDB):** This is where all the application data is stored. It's a flexible, non-relational database that holds information about user profiles, property listings, rental applications, and payments.

****

**Table-1 : Components & Technologies:**

|  |  |  |  |
| --- | --- | --- | --- |
| **S.No** | **Component** | **Description** | **Technology** |
|  |  |  |  |
|  | User Interface | How user interacts with application e.g.  Web UI, Mobile App, Chatbot etc. | HTML, CSS, JavaScript / React Js etc. |
|  | Application Logic-1 | Logic for a process in the application | javascript |
|  | Application Logic-2 | Logic for a process in the application | Renting services across world |
|  | Application Logic-3 | Logic for a process in the application | House rent logic |
|  | Database | Data Type, Configurations etc. | NoSQL, |
|  | Cloud Database | Database Service on Cloud | No use of this |
|  | File Storage | File storage requirements | Basic system storage |
|  | External API-1 | Purpose of External API used in the application | Working properly as listing any property |
|  | External API-2 | Purpose of External API used in the application | Showing result on dashboard |
|  | Machine Learning Model | Purpose of Machine Learning Model | No use of ML |
|  | Infrastructure (Server / Cloud) | Application Deployment on Local System / Cloud  Local Server Configuration:  Cloud Server Configuration : | Local |

**Table-2: Application Characteristics:**

| **S.No** | **Characteristics** | **Description** | **Technology** |
| --- | --- | --- | --- |
|  | Open-Source Frameworks | Resources provided by course guide and chatgpt and AI | Technology of Opensource framework |
|  | Security Implementations | Use secure network to access and implement the project. | - |
|  | Scalable Architecture | 3 tier architecture | Fontend and backend mern stack |
|  | Availability | More available for every user | databases |
|  | Performance | Impactful performance | Implementation plateforms |